



- 1 -

1 A METHOD FOR CREATING AND MAINTAINING WORLDWIDE E-COMMERCE

2

3 BACKGROUND OF INVENTION

4 1. Field of the Invention

5 The field of this invention relates to a virtual network of e-commerce e-malls, satellite e-  
6 malls, e-shops, e-distributors and web sites. More specifically, the present invention relates  
7 means for creating and maintaining worldwide e-malls and each of these e-malls will offer  
8 means for creating e-shops, e-distributors and web sites without the need of an e-commerce  
9 infrastructure or even a web server.

10

11 2. Prior Art

12 The Internet has a tremendous potential with its worldwide reach; also, there are a lot of  
13 challenges and opportunities. At the present, there are needs for easy and affordable  
14 worldwide e-commerce solutions where seller can have their goods and services sold without  
15 the expertise or the expenses that today's e-commerce requires.

16

17 Today's e-commerce web sites henceforth called e-shop(s) are of a dynamic type with  
18 products and/or services that are available to a broad base of buyers. One good example of a  
19 dynamic e-shop is Amazon.com.

1  
2 One other type of e-commerce setup is the e-shopping mall where dynamic e-shops are  
3 created and updated directly by a user and henceforth called e-mall(s). The four most popular  
4 are: VStore.com, ViaWeb.com, Bcentral.com and BigStep.com.

5  
6 Stores in these e-malls are treated as independent dynamic e-shops with specific URL  
7 (Uniform Resource Locator) addresses and their products/services are only available within  
8 their closed environment. Thus, products/services cannot be shared among other e-malls or e-  
9 shops even within their own network of dynamic e-shops at the e-mall.  
10 Except VStore.com where all e-shop virtually sells products from distributors.

11  
12 The dynamic e-mall setup does not enhance the shopping experience nor facilitates the  
13 interaction between buyer and seller. Since a buyer will have to move from e-shop to e-shop  
14 in the e-mall. Time is thus wasted and sales can be lost. Furthermore, the dynamic e-mall  
15 concept cannot be created without an elaborate and expensive e-commerce infrastructure that  
16 requires extensive knowledge and expertise.

17  
18 Currently, dynamic e-mall will not allow the creation of specialized e-shops that can sell  
19 their products/services in conjunction with similar products/services from others e-shops.

1

2       Today's e-commerce requires solutions where seller can have their products/services  
3 available to a broad base of buyers, also, virtually available to other e-shops, satellite e-malls  
4 and e-malls where they will be offered to a broader clientele base. For this to be possible, the  
5 process for creating and updating e-malls, satellite e-malls, e-shops, e-distributors and web  
6 sites must be on-line and easy to setup and use.

7

8       Buyers on the other hand, need a solution where they will have a broad selection without  
9 having to go to many different e-shops to find what they're looking for, and also be able to  
10 view web pages in their own native language.

11

12       It is the object of this invention to offer easy and affordable e-commerce solution  
13 worldwide with a single e-commerce and e-services resource infrastructure as to allow anyone  
14 anywhere in the world to provide e-commerce solution without the need of an e-commerce  
15 infrastructure or even an Internet server.

16

## 1 SUMMARY OF THE INVENTION

2 It is the object of this invention to demonstrate a virtual electronic shopping mall where  
3 on-line users can create and update e-malls which in turn offers others the ability to host e-  
4 shops and web sites offering products/services. Also, an e-mall will have means for  
5 customizing a satellite e-mall and means for hosting pre-set ones. Moreover, this invention  
6 will allow creating and maintaining of the dynamic e-mall concept without the need of an e-  
7 commerce infrastructure or even the need of a web server.

8  
9 It will further have the ability to exchange interfaces uploaded by a user or allow the user  
10 to select one provided by the virtual network (the host computer with the technology  
11 infrastructure).

12  
13 Also, the ability to display interfaces in more than one foreign language for: e-malls,  
14 satellite e-malls, e-shops, e-distributors and web sites. All e-malls, satellite e-malls, e-shops,  
15 e-distributors and web sites are within a network where products and services can be shared  
16 among any e-mall, satellite e-malls, e-shop or web site by making them virtual.

17  
18 For instance, an e-shop can sell its products dynamically, or have other e-malls; satellite  
19 e-malls and e-shops in the virtual network selling them virtually. The advantage of this virtual

1 electronic network environment is to make products and services available to a broader base  
2 for both, sellers and buyers.

3  
4 The virtual electronic network environment will enhance the way people shop  
5 electronically. Thus, making a wider selection of products and services available to any e-mall,  
6 satellite e-mall, e-shop and web site within the virtual network. It will offer means for the  
7 creation of specialized e-shops, satellite e-malls, e-malls and web sites, tailored uniquely to a  
8 specific market segment. Further, it will simplify buyers' decision by offering them a broad and  
9 specialized selection of products/services.

10  
11 In one other embodiment means for having one content page's object used to search  
12 other contents at the virtual server and fetch at least one other content, also, having means to  
13 present a single page having a plurality of contents from two or more sources.

14  
15 Yet in another embodiment means to track users viewing contents and create a history of  
16 visited contents for each user.

17  
18 Still in one other embodiment will have means to register user and present content based  
19 on each registered user's preset preferences.

1

2        Yet another embodiment will have means to present content based on a user's previously  
3 viewed content page.

4

5        Also in another embodiment will have means for satellite e-mails with specialized contents  
6 to be integrated into e-mails and web sites.

7

8        In one other embodiment web sites will be able to present content virtually and the  
9 contents they are presenting are hosted by other web sites

10

11       Still in another embodiment it shows means for exchanging interface and preset interface  
12 rendering from a disk-file parameters.

13

14       It is also the intent of this invention to permit users to include e-commerce at web sites  
15 that are already in use. Thus, offering a new way to enhance them without the cost of  
16 providing the e-commerce infrastructure that is required, or even having to install and  
17 maintain a web server.

18

1 It will further permit the creation of virtual web sites, other than shopping. For instance, a  
2 web site may be virtual on-line schools, cities, etc. The use of this invention is without limits,  
3 and, as more goods/services are offered, the more it will allow the creation of a virtual  
4 shopping and non-shopping web sites and a combination of both.

5  
6 It will also integrate the world by allowing an e-shop in one country to sell in another  
7 country or to cross-sell products from a distributor or a manufacturer and having them  
8 delivered directly to the buyer, thus reducing storage, shipping and handling costs.

9  
10  
11 **BRIEF DESCRIPTION OF THE DRAWINGS**

12 The accompanying drawings, which are incorporated in the form a part of this  
13 specification, illustrate embodiments of the invention and, together with the description, serve  
14 to explain the principles of the invention:

15  
16 Fig.1 illustrates a worldwide e-commerce sharing a single e-commerce resource  
17 infrastructure.

1        Fig. 2 illustrates the e-commerce resource infrastructure available to each e-commerce of  
2        Fig. 1.

3

4        Fig.3 illustrates the virtual e-shopping network system where e-malls, e-shops, e-  
5        distributors and web sites share a single resource.

6

7        Fig. 4 illustrates a communication link between a client and a server computer. The  
8        process of converting ASP pages into HTML by the server computer and transmit it to the  
9        client computer thereafter.

10

11       Fig. 5 illustrates the process that takes place in converting ASP pages templates into  
12       HTML web pages.

13

14       Fig. 6 illustrates a dynamic electronic store.

15

16       Fig. 7 illustrates a dynamic electronic shopping mall.

17

18       Fig. 8 illustrates multiple dynamic electronic shopping mall with multiple dynamic  
19       electronic shops.



1  
2        Fig. 9 illustrates how dynamic electronic shops become virtual ones in the dynamic  
3 shopping mall.

4  
5        Fig. 10 illustrates virtual electronic shopping mall with dynamic and virtual electronic  
6 shops.

7  
8        Fig. 11 illustrates multiple dynamic electronic shops with departments, category, sub-  
9 category and products in a dynamic shopping mall.

10  
11       Fig. 12 illustrates how a dynamic electronic shop's departments, category, sub-category  
12 and products become virtual ones in the electronic dynamic shopping malls and dynamic  
13 electronic stores.

14  
15       Fig. 13 illustrates a dynamic database table that represents the illustration of Fig. 11.

16  
17       Fig. 14 illustrates a virtual database table that represents the illustration of Fig. 12.

18

1        Fig. 15 illustrates multiple dynamic servers' setup sharing database objects with the virtual  
2 server.

3

4        Fig. 16 illustrates a database table to be used to translate web page text objects into  
5 multiple foreign languages.

6

7        Fig. 17 illustrates a web page drop down elements.

8

9        Fig. 18 illustrates the source code for the web page drop down of Fig. 17.

10

11       Fig. 19 illustrates an ASP program to be used to translate web page database text objects.

12

13       Fig. 20 illustrates a database table with products in multiple foreign languages.

14

15       Fig. 21 illustrates a database table with database table's object in multiple foreign  
16 languages.

17

18       Fig. 22 illustrates a client and a server computer's setup used by the server computer to  
19 process ASP pages, and transmits its result in the HTML format to a client computer.

1

2 Fig. 23 illustrates a web page template.

3

4 Fig. 24 illustrates a web page template with include files.

5

6 Fig. 25 illustrates the process of exchanging a web page template.

7

8 Fig. 26 illustrates a web page template with include files and their respective folders.

9

10 Fig. 27 illustrates folders for the include files of Fig. 26.

11

12 Fig. 28 illustrates a client receiving an interface from multiple servers.

13

14 Fig. 29 illustrates a method of uploading a file with settings to the virtual network server.

15

16 Fig. 29a illustrates two files with settings in a folder for each template.

17

18 Fig. 30 illustrates a file with settings applied to a web page.

19

1 Fig. 31 illustrates virtual network's satellite e-malls and e-malls' satellite e-malls.

2

3 Fig. 31a illustrates satellite e-malls receiving e-services through the virtual network.

4

5 Fig. 31b illustrates two satellite e-malls receiving e-services through the virtual network  
6 from three different sources.

7

8 Fig. 31c is a further embodiment of fig. 31b and it illustrates two satellite e-malls and two  
9 e-malls virtually presenting a single content page from satellite e-malls and VNRI.

10

11 Fig. 31d illustrates a content page created by using contents from two separate sources:  
12 one from VNRI and another from a remote computer.

13

14 Fig. 31e is a further embodiment of fig. 31d and it illustrates three methods of using  
15 HTML objects on a content page for allowing the fetching of contents from a remote  
16 computer and VNRI.

17

18 Fig. 31f illustrates contents from a remote computer routed through VNRI to another  
19 computer that supplies them virtually to a client computer.

1  
2 Fig. 32 illustrates a web page divided in two windows, one for e-commerce and the other  
3 for e-services.

4  
5 Fig 33 illustrates the web page of Fig. 32 with e-commerce on the left and e-services on  
6 the right.

7  
8 Fig. 34 illustrates a list of products for the chosen sub-category.

9  
10 Fig. 34a is a further embodiment of fig. 34 and it illustrates a content page created by  
11 using contents from two separate content sources within VNRI.

12  
13 Fig. 35 illustrates a method of user tracking between a web server and a web browser.

14  
15 Fig. 36 illustrates a web browser's user viewing web pages from multiple web sites and e-  
16 shops.

17  
18 Fig. 37 illustrates a method of a web server tracking user surfing experience.

19

1 Fig. 38 illustrates a method of a web server presenting user's surf list for review.

2  
3 Fig. 39 illustrates a user reviewing web pages from his/her surf list.

4  
5 Fig. 40 illustrates the managing of user's customized contents.

6  
7 Fig. 41 illustrates a user viewing previously viewed customized contents.

8  
9  
10 **DESCRIPTION OF THE PREFERRED EMBODIMENTS**

11 The present invention now will be described more fully hereinafter with reference to the  
12 accompanying drawings, in which preferred embodiments of the invention are shown. This  
13 invention may, however, be embodied in many different forms and should not be construed as  
14 limited to the embodiments set forth herein. Rather, these embodiments are provided so that  
15 this disclosure will be thorough and complete, and will fully convey the scope of the invention  
16 to those skilled in the art. Like numbers refer to like elements throughout.

17  
18 As will be appreciated by one of skill in the art, the present invention may be embodied as  
19 a method, a virtual network, or a computer program product. Accordingly, the present

1 invention may take a form of an entirely software embodiment or an embodiment combining  
2 software and hardware. Furthermore, the present invention may take the form of a computer  
3 program product on a computer-readable storage medium having computer-readable program  
4 code means embodied in the medium. Any computer readable medium may be utilized  
5 including hard disks, CD-ROMs, optical storage devices, or magnetic devices.

6  
7 As is understood by those skilled in the art of Web client/server communications, a user  
8 access a server by establishing a TCP connection between client and server. Client and server  
9 communicate by using HTTP protocol over a TCP connection. Data transferred from servers  
10 to clients are HTTP objects (e.g. HTML objects).

11  
12 Furthermore, any reference to names of a product or of a company is for the purpose of  
13 clarifying our discussion and they are registered to their respective owners.

#### 14 15 GLOSSARY OF USED TERMS

16 Before explaining this invention, let us first explain some of the terms that will be used  
17 throughout.

1        **ASP** stands for *Active Server Page* and it is used to dynamically create web pages on the  
2 server side and transmit them to a requesting client as HTML objects.

3  
4        **Back end** means a computer system where users retrieve (content) from and view it at  
5 another computer (front end). In the case of the Internet a computer with a web server is the  
6 back end.

7  
8        **Cookie** is a file kept by a web browser at the end user computer where the server or a  
9 script program running at the web browser send requests to the browser to save a value in the  
10 cookie file for later retrieval. It has a cookie name and the cookie's value and the date that it  
11 will expire.

12  
13        **Dynamic e-shop** and **dynamic web site** is when products (also called goods) and  
14 services are part of the hosting e-shop or web site. That is, the e-shop or web site will do all  
15 the required maintenance. Each e-shop or web site is hosted within an e-mall.

16  
17        **E-services** are services (also called contents) supplied to VNRI by a third party or part of  
18 VNRI. E-services are used in conjunction with VNRI's e-commerce. These e-services have  
19 means for associating with the e-commerce counterpart.



1       **Front end** means a computer system where users view information (also called content  
2 and used interchangeably) that is located at another computer system (back end). In the case  
3 of the Internet a computer with a web browser is the front end.

4  
5       **HTML** stands for Hypertext Markup Language.

6  
7       **HTTP** stands for Hypertext Transfer Protocol.

8  
9       **JavaScript** is a commonly used language by clients and servers as well. On the client side  
10 it is used as means of interacting with HTML objects. On the server side it is used as a script  
11 language and it works in conjunction with the ASP and others technologies.

12  
13       **Satellite e-mail** is a sub-set of an e-mail created by the virtual network system or by each  
14 individual e-mail (in the case of customized satellite e-mail). An e-mail has the option to  
15 include or to exclude services from a satellite e-mail.

16  
17       **Session Variable** is a variable that the web server keeps in its memory for the duration of  
18 a web browser user visit to the web site or, in most cases, 20 minutes after the user access the

1 last page from the web server. Any value can be assigned to a session variable. One good  
2 example is the use of session variables with the ASP technology.

3  
4 **SQL** stands for *Structured Query Language* and it is a text string used by a database  
5 server (a specialized software to manage databases) as to guide it in querying and retrieving  
6 database objects thereof.

7  
8 **TCP** stands for Transfer Control Protocol.

9  
10 **URL** stands for *Uniform Resource Locator* and it is an Internet address used by a web  
11 browser to fetch a web page object from a web server.

12  
13 **WEB SERVER** is a computer with specialized software to manage communication  
14 between a client's web browser and the server computer. Also, it communicates with others  
15 technologies that are within the server computer.

16  
17 **VNRI** stands for *Virtual Network Resource Infrastructure* (also called virtual network,  
18 virtual server and used here interchangeably) and it is the hosting environment hosting all e-  
19 malls, satellite e-malls, e-shops, e-distributors and web sites dynamically and virtually. Also, it

1 is where the e-commerce infrastructure is hosted (hardware, software and the Internet  
2 gateway).

3  
4 **Virtual e-mall** is when an e-mall place goods/services from any e-shop or web site from  
5 the virtual network. Products offered by a virtual e-mall can be virtual ones and/or they can  
6 dynamic as well and hosted by the virtual e-mall.

7  
8 **Virtual e-shop or virtual web site** is used when an e-shop or web site offers  
9 products/services from another e-shop or web site within the virtual network.  
10 Products/services are called **virtual** products because they are available to an e-shop or web  
11 site other than the hosting one.

12  
13  
14 **I) THE VIRTUAL NETWORK**

15  
16 Let us further explore the virtual network. There will be many e-malls, e-shops, e-  
17 distributors and web sites in the virtual network. An e-mall may or may not have a dynamic e-  
18 shop or web site. Also, an e-shop or web site may or may not have products and/or services.  
19 An e-mall can have an e-shop or web site that is empty (without products or services). Based

1 on the previous situation, all products and/or services offered by an empty e-shop or web site  
2 are virtual ones.

3  
4 Lets further explore the interaction that goes between a client (user) computer and the  
5 virtual network server.

6  
7 A user at a client web browser initiates communication with the virtual network server  
8 and register an e-mall. Other users will do the same. After an e-mall is registered it is ready to  
9 receive e-shops or web sites.

10  
11 Let us say that, a user registers an e-shop and call it "E-SHOP A". The user of *E-SHOP*  
12 *A* then adds products to it. Other registered e-shops at the network will place products in their  
13 e-shops as well. This process is called dynamic, since products are within each individual e-  
14 shop.

15  
16 For the sake of our discussion, lets assume that, there are two e-shops: **E-SHOP A** and  
17 **E-SHOP B**. They both have products. Now the user of **E-SHOP B** accesses the virtual  
18 network server, receives a page with information about services, products, departments,  
19 categories, sub-categories, e-shops and web sites. This user now includes, *products* from **E-**

1    **SHOP A** and a *sub-category*, then, sends it to the virtual network server. The network server  
2    receives and saves **E-SHOP B** selection (products from **E-SHOP A** and a sub-category) into  
3    a virtual database table.

4

5        Now, a user accessing **E-SHOP B** will have products from **E-SHOP A** and a sub-  
6    category with all of its products (virtually), also, all of **E-SHOP B** products (dynamically).

7

8        As we've said before, a client computer communicates with a server computer over the  
9    Internet or Intranet by a TCP connection. Moreover, a client sends HTTP requests to server.  
10   The server then process such requests and creates a web page (e.g. HTML objects), and sends  
11   it back to the client.

12

13        Furthermore, a client computer has a screen that is the means of interfacing with a user. It  
14   also has web browser software that runs in the client computer memory. The web browser is  
15   the client computer user's interface. And it presents a web page as objects (e. g. HTML  
16   objects) to the user. The objects can be text, audio, video, image, forms, links, etc. Also, the  
17   web browser communicates with a server computer.

18

1       Let's move on and discuss the Virtual Network Resource Infrastructure (henceforth called  
2       VNRI). The virtual network offers an infrastructure for worldwide e-commerce at a single  
3       location and available to anyone without any e-commerce infrastructure or even a server.

4  
5       This VNRI infrastructure is located at a single location and it will be the back-end for  
6       worldwide front-ends accessing it. Also, this infrastructure will include software programs to  
7       manage all the VNRI associated e-malls, satellite e-malls, e-distributors and web sites.  
8       Further, it will include Internet, e-mail, credit cards gateways and all the necessary hardware.

9  
10      The objective of this invention is to allow the creation of a plurality of e-malls and each e-  
11      mall will use VNRI's hosting and management means and offer e-shops, e-distributors and  
12      web site an e-commerce and e-service infrastructure (hosting and management) without  
13      having the infrastructure of their own. Each e-mall will be able to sell its e-shops products and  
14      e-shop's products located in other e-malls. Also, an e-shop will be able to sell its products and  
15      others e-shops' products. There will be only a single infrastructure for a worldwide use and it  
16      will be within VNRI. As it is now clear, e-malls, satellite e-malls, e-distributors and web sites,  
17      all use the VNRI for e-commerce and e-services. This process will give the idea that each one  
18      has an e-commerce and/or e-service infrastructure of its own because each will offer an end-  
19      user means for using all the available resources within the VNRI.

Fig. 1 shows e-commerce located at different parts of the world. Fig. 2 shows the e-commerce infrastructure that is available to each e-commerce of Fig. 1. Figs. 23, 24, 25, 26, 27 and 28 show the process of uploading interfaces to each of the e-commerce of Fig. 1. Each e-commerce of Fig. 1 is able to present its solutions as if each one had its own e-commerce infrastructure.

Back to Fig. 1. It shows a method of using a single e-commerce resource infrastructure at a single location and providing e-commerce solution worldwide without each one having to replicate the e-commerce resource infrastructure at each location.

As it is shown, there is an e-commerce located in Brazil 102, USA 104, Mexico 106, France 108, Germany 110 and Canada 112. Although the e-commerce is spread worldwide, the e-commerce resource infrastructure is at a single location at VNRI 100.

Fig. 2 is the e-commerce infrastructure that is available to each e-commerce that was presented in Fig. 1 and it is within VNRI 100 of Fig. 1.

1        There are three layers of e-commerce within VNRI - Fig. 2. E-distributors/e-  
2        manufacturers 260 (henceforth called e-distributors). E-malls 268 and satellite e-malls (pre-set  
3        satellite e-mall 271 and customized satellite e-mall 273).

4  
5        Each will present e-commerce differently. Each e-distributors 260 has an e-commerce  
6        web site and will be able to sell its products directly (e-shop A 262) or have other e-shops (e-  
7        shop B 281 - arrow line 282) and e-malls (e-mall 268 - arrow line 266) selling them virtually.  
8        In either case, end user/shoppers (henceforth called end user) will be able to view each one  
9        independently. End user A 278 views e-shop A 262 and e-shop B 281.

10  
11       E-malls, on the other hand, have the means to allow the creation of e-commerce e-shops  
12       or use the services of agents. Each e-shop will be able to sell its products directly or have  
13       them sold by the e-mall and its affiliated satellite e-malls. E-mall 268 has e-shop B 281 (arrow  
14       line 265). As shown, e-shop B 281 has end user A 278 viewing its products and also end user  
15       B 276 is viewing them through e-mall 268, pre-set satellite e-mall 271 and customized satellite  
16       e-mall 273.

17  
18       As presented, all products of an e-shop can be sold by the e-shop, by the e-mall where e-  
19       shop is dynamically located and by the e-mall's affiliated satellite e-malls. E-mall 268 has e-



1 shop B 281 (arrow line 265). E-mall 268 sells e-shop B 281 products (arrow line 267). E-mall  
2 268 makes all of its e-shops products available to each of its affiliated satellite e-malls. As  
3 shown, e-mall 268 has a pre-set satellite e-mall 271 and a customized satellite e-mall 273. And  
4 each one sells products from e-shop B 281 (arrow line 279).

5

6 Also, an e-mall will have the means as to allow the use of agents' services. An e-mall can  
7 have one or more agents and each agent will be able to offer their expertise to e-shops in a  
8 one-to-one basis. An e-shop can be created directly to an e-mall or by an agent, also, an agent  
9 can manage e-shops, e-distributors and web sites for one or more e-malls. E-mall 268 has  
10 agent 286 and it is the creator (arrow line 288) of e-shop B 281. For sake of simplicity e-shop  
11 B 281 has e-mall 268 point to it (arrow line 265) and Agent 286 (arrow 288). In this case it  
12 means that Agent 286 created e-shop B 281 (arrow 288) by using the e-mall 268 as the holder  
13 of e-shop B 281 (arrow 265).

14

15 Before proceeding any further, lets explain what an agent is. An agent is a person  
16 authorized to provide management services to the e-mall's e-shops. Also, an agent has log in  
17 and management rights to log in and manage e-shops for any e-mall in the virtual network  
18 resource infrastructure. An e-shop can be created and managed by its owner or have an agent  
19 to represent the e-shops within the e-mall.

1  
2        Fig. 3 shows a virtual e-commerce environment where e-malls (302, 303, 304, and 305)  
3        with e-shops share the virtual network resources 301 (e-commerce infrastructure: hardware,  
4        software and the Internet gateway). It further shows the ability for one e-shops to cross-sell  
5        other e-shops' products virtually (306, 307, 308 and 309).

6  
7        As it will be understood by those skilled in the art. The virtual network resources will  
8        have all the required software and hardware that is needed to host e-malls, satellite e-malls, e-  
9        shops, e-distributors and web sites over the Internet or over a computer network.

10  
11        As it can be seen at Fig. 3, an e-mall will be able to host e-shops as if they had their own  
12        e-commerce. And since the system offers means for user plug in interfaces and the availability  
13        of different foreign languages, anyone anywhere in the world will be able to offer e-commerce  
14        services without the expenses involved.

15  
16        Fig. 4 shows a communication method between a client computer and a server computer.  
17        It should be understood that, not all parts of the client and server computers are shown in the  
18        drawing. It is done as is for sake of simplicity, and it is not intended to hide or obscure this

1 invention. Furthermore, those skilled in the art will be able to follow the embodiments and  
2 fully understand its meanings.

3  
4 For example, a memory is present at the client computer, since, this is where the web  
5 browser software resides and is executed. The same is true for the screen. Where the web  
6 browser presents web objects in a visual form for the user. Neither a means for saving  
7 information like a disk unit nor a means for entering data such as a keyboard or a mouse are  
8 shown. On the server side, a memory (where Web Server and ASP server are resident) and a  
9 storage unit (where the Database, ASP Programs and Web Page Template are stored) are  
10 shown. It can also have a screen, mouse, keyboard, etc.

11  
12 Back to Fig. 4. Client computer 410 establishes a communication link 420 with a server  
13 computer 440. The client computer 410 represents the computer with a web browser where  
14 an end user views web sites. Server computer 440 is the VNRI hosting the e-commerce.

15  
16 A user at client computer 410 enters requests through the user interface 411 (web  
17 browser). These requests are processed at the web browser 412 that is resident in the client  
18 computer memory. For instance, if a link to another page is selected by the end user. The web  
19 browser 412 will establish a communication link 420 with server computer 440 and transmit a

1 request for the new web page. Server computer 440 receives the request through the  
2 communication link 420 and passes it to the resident web server software 430.  
3

4 After the server computer 440 receives the request for a new web page, let's say that, the  
5 requested web page is an ASP page. The web server 430 connects with the ASP server 431. It  
6 in turns reads the web page template 436 and ASP programs 437. Next, the ASP server 431  
7 will include the ASP programs 437 into the web page template 436, thus, forming a single file  
8 and executes it. There are cases that as the file is executed, the ASP program will need data  
9 from a database. In such cases, the ASP server 431 establishes a link 432 with the database  
10 433 then reads/writes to it, as needed.  
11

12 After a line of programming code of the ASP program 437 is executed, its result (if any)  
13 is inserted into the web page template 436, thus, creating a final web page (more of it later).  
14 After all lines of the ASP program 437 are executed, the ASP server 431 will pass the final  
15 web page to the web server 430. The web server 430 establishes a communication link 420  
16 with client computer 410 and transmits the web page to the web browser 412. And the new  
17 web page is displayed on the user interface (screen) 411. Thus finalizing the communication  
18 process between a client and a web server computer.  
19

1        Fig. 5 further explains the ASP process. ASP template 551 receives ASP programs A, B,  
2        C and D 552 (usually, they are *include files*). Next, a web page 550 in the HTML format is  
3        created. It is now clear to those skilled in art how the communication process between a client  
4        computer and a server computer happens. Also, the process of creating web pages at the  
5        server computer with the use of ASP technology.

6  
7        Since the object of this invention is to present the process of creating worldwide virtual e-  
8        malls and each one offering e-commerce to e-shops, e-distributors and web sites, lets first  
9        explain the dynamic process involved with e-shops and e-malls. Next, the process that is  
10       involved with virtual e-malls and e-shops will be presented as well.

11  
12       Fig. 6 shows a dynamic e-shop with departments. Each department having a category, a  
13       category having a sub-category, and the sub-category having products. It should be noted  
14       that, a department can have one or more categories, a category can have one or more sub-  
15       categories, and a sub-category can have one or more products. Fig. 7 shows a dynamic E-  
16       **MALL 760** with three e-shops: **E-SHOP A 761**, **E-SHOP B 762** and **E-SHOP C 763**. A  
17       dynamic e-mall is the host for one or more dynamic e-shops (Fig. 6).

18

1        Now that we know the dynamic process, let's move on and explore our virtual network  
2        concept. Fig. 7 shows an e-mall 760 with three e-shops: E-shop A 761, E-shop B 762 and E-  
3        shop C 763. Fig. 1 shows the VNRI with e-commerce worldwide. Each e-commerce at Fig. 1  
4        will have the same e-commerce infrastructure that is available to the e-mall of Fig. 7. It is like  
5        we're taking a dynamic e-mall of Fig. 7 with dynamic e-shop of Fig. 6 and offering means for  
6        their replication worldwide. Also, each one will have the complete e-commerce infrastructure  
7        that is available at the VNRI (Fig. 2).

8  
9        Furthermore, the VNRI has hosting and management means for hosting and managing  
10       various e-malls and each e-mall will use the VNRI hosting and management means for hosting  
11       and managing e-shops, e-distributors and web sites. The VNRI has the means for allowing the  
12       offering of e-commerce solution to e-malls, satellite e-malls, e-shops, e-distributors and web  
13       sites.

14  
15       Fig. 8 is an overview of a virtual e-mall where four e-malls coexist and each having their  
16       dynamic e-shops. As we further explore it, we'll see that each e-shop is able to share its  
17       products with others e-shops and e-malls in the virtual network. Each e-mall has three  
18       dynamic e-shops. E-MALL A 800 has E-SHOP A-1 806, E-SHOP A-2 805 and E-SHOP A-3  
19       804. E-MALL B 801 has E-SHOP B-1 815, E-SHOP B-2 814 and E-SHOP B-3 813. E-

1 MALL C 803 has E-SHOP C-1 812, E-SHOP C-2 811 and E-SHOP C-3 810. E-MALL D  
2 802 has E-SHOP D-1 807, E-SHOP D-2 808 and E-SHOP D-3 809.

3

4 Fig. 9 is a further embodiment of Fig. 8. It shows how to create a virtual e-shop. Dynamic  
5 e-shops are made virtual once they are placed at others e-malls or e-shops within the VNRI.  
6 For instance, dynamic E-SHOP A-1 806 is now virtual 825 at E-MALL D 802. Dynamic E-  
7 SHOP A-3 804 is now virtual 820 at E-MALL B 801. Dynamic E-SHOP B-3 813 is now  
8 virtual 821 at E-MALL C 803. Dynamic E-SHOP D-2 808 is now virtual 823 at E-MALL C  
9 803, and also, virtual 824 at E-MALL A 800. Finally, dynamic E-SHOP D-3 809 is virtual  
10 822 at E-MALL C 803.

11

12 Fig. 10 is a further embodiment of Fig. 9 with virtual e-shops included. As it is shown, E-  
13 SHOP A-1 is dynamic 806 at E-MALL A 800 and it is virtual 825 at E-MALL D 802. E-  
14 SHOP A-3 is dynamic 804 at E-MALL A 800 and it is virtual 820 at E-MALL B 801. E-  
15 SHOP B-3 is dynamic 813 at E-MALL B 801 and it is virtual 821 at E-MALL C 803. E-  
16 SHOP D-3 is dynamic 809 at E-Mall D 802 and it is virtual 822 at E-MALL C 803. E-SHOP  
17 D-2 is dynamic 808 at E-MALL D 802 and it is virtual 823 at E-MALL C 803, and also,  
18 virtual 824 at E-MALL A 800.

19

Further, a dynamic e-shop becomes a virtual one by its products being shared by others e-shops in the virtual network. Let's further explain it. If a buyer is accessing E-MALL A 800, he/she will have all dynamic e-shops (E-SHOP A-1 806, E-SHOP A-2 805 and E-SHOP A-3 804) and the virtual E-SHOP D-2 824 as well. E-MALL D 802, which has E-SHOP A-1 825 now available as a virtual one, and it is dynamic 806 at E-MALL A 800. E-MALL C 803 has E-SHOP D-2 823 and E-SHOP B-3 821 as virtual, and finally, E-MALL B 801 has E-SHOP A-3 820.

Fig. 11 Shows dynamic departments, categories, sub-categories and products within individual e-shops, as we will see, they will also become virtual ones, like we've seen with the e-shops. Department, category, sub-category and products are dynamic at E-SHOP A-2 1105 at E-MALL A 1100. The same is true with E-SHOP C-2 1111 at E-MALL C 1103. E-SHOP B-1 1115 and E-SHOP B-3 1113 both are dynamic at E-MALL B 1101.

For sake of simplicity, only one department with one category and one sub-category are shown. It should be noted that, more than one department can exist at an e-shop; a department can have one or more categories, and a category can have one or more sub-categories.



Fig. 12 is a further embodiment of Fig. 11. It shows dynamic departments, categories, sub-categories, products and e-shops becoming virtual ones. As it is shown, E-SHOP A-2's 1105 (Department A-2 1140, Category A-2 1141 and Sub-category A-2 1142) are now virtual 1150 at E-MALL C 1103. E-SHOP C-2's 1111 (Product C-2-1 1160, Product C-2-2 1161 and Product C-2-3 1162) are now virtual 1151 at E-MALL A 1100. E-SHOP A-2 1105 is virtual 1153 at E-MALL B 1101. E-SHOP B-3 1113 is virtual 1152 at E-MALL C 1103 and E-SHOP B-1 1115 is virtual 1154 at E-SHOP B-3 1113. As it can be seen, dynamic e-malls and e-shops can have any combination of e-shops, departments, category, sub-category and products virtually.

Before we continue with our discussion, let's first review the dynamic database table of Fig. 13 and virtual database table of Fig. 14. The dynamic database table of Fig. 13 holds all data information about all e-malls e-shops and web sites. Virtual database table of Fig. 14 is the database table that represents the virtual part of the virtual network.

Fig. 13 shows a database table for a dynamic configuration of Fig. 11. Let's go back to Fig. 11 and explain E-SHOP A-2 1105. As it is shown, Department A-2 1140, Category A-2 1141, Sub-category A-2 1142, Product A-2-1 1143, Product A-2-2 1144 and Product A-2-3 1145 are within E-SHOP A-2 1105 located at E-MALL A 1100. The first three rows (rows 1, 2 and 3 of the ID column) of the dynamic database table of Fig. 13 shows, **E-Shop A-2** at

1 the column **Shop\_Name** it represents the E-SHOP A-2 1105 (Fig. 11). **Department A-2** at  
2 the column **Department** it represents Department A-2 1140 (Fig. 11). **E-Mall A** at the  
3 column **Mall\_Name** it represents E-MALL A 1100 (Fig. 11). **Category A-2** at the column  
4 **Category** it represents Category A-2 1141 (Fig. 11). **Sub-category A-2** at the column **Sub-**  
5 **category** it represents Sub-category A-2 1142 (Fig. 11). **Product A-2-1, Product A-2-2** and  
6 **Product A-2-3** at the column **Product\_ID** they represent Product A-2-1 1143, Product A-2-  
7 2 1144 and Product A-2-3 1145 (Fig. 11). Anyone skilled in the art will be able to follow the  
8 remaining rows of the table of Fig. 13 and the diagram of Fig.11.

9 Fig. 14 shows a database table for the virtual part of Fig. 12. And as we've done with Fig.  
10 13, we'll explain the first three rows (rows 1,2 and 3 of the ID column). The column  
11 **Virtual\_Mall** represents the e-mall that virtually receives products/services from other e-  
12 shops. **E-MALL A** at the column **Virtual\_Mall** represents E-MALL A 1100 (Fig. 12). The  
13 value *Product* at the column **Type** of Fig. 14 says that they are virtual products 1151 (Fig.  
14 12). And finally, the names of the virtual products at the column **Virtual\_Type\_Name**  
15 (Product C-2-1, Product C-2-2 and Product C-2-3) represent products (Product C-2-1 1160,  
16 Product C-2-2 1161 and Product C-2-3 1162 - Fig. 12). Also, *Department*, *Category Sub-*  
17 *category* and *Shop* at the column **Type** of Fig. 14 indicate their respective dynamic types of  
18 Fig. 12.

1       It should be noted that, dynamic database table of Fig. 13 and virtual database table of  
2       Fig. 14 can be related, although not shown. Also, anyone skilled in the art will be able to use  
3       both tables and formulate SQL's text string to create queries that will reflect a single resultant  
4       table. A combination of SQL text string that represents dynamic table (Fig. 13) and retrieved  
5       database objects from the virtual table (Fig. 12) will be used to form a single SQL text string to  
6       query the dynamic database table (Fig. 13) and, retrieve a single database table.

7       Let's go back to Fig. 12 and use E-MALL B 1101 for our next example. E-MALL B  
8       1101 has E-SHOP B-1 1115 and E-SHOP B-3 1113 (dynamic e-shops). It also has E-SHOP  
9       A-2 1105 as a virtual e-shop 1153. Now we need to create a single database table from the  
10      above example. Two searches are needed. One at the dynamic table (Fig. 13) and another at  
11      the virtual table (Fig. 14). At the dynamic table of Fig. 13, a search at the column **Mall\_Name**  
12      will be conducted and all e-shops for E-MALL B will be retrieved (rows 4 through 9 of the ID  
13      column). Next, at the virtual table of Fig. 14 a search at the column **Virtual\_Mall** is also  
14      conducted and it will search for E-MALL B (it is the e-mall that has the virtual E-SHOP A-2  
15      and it is located at **Virtual\_Type\_Name** - row # 4 of the **ID** column). Next, E-SHOP A-2  
16      will be retrieved from the column **Virtual\_Type\_Name**. The first part of the query retrieves  
17      the dynamic part and the second one retrieves the virtual part. Following is a SQL query to do  
18      just that.

```
1 -----  
2 SELECT * FROM DynamicTable WHERE Mall_Name='E-MALL B' OR Shop_Name IN  
3 (SELECT Virtual_Type_Name FROM VirtualTable WHERE Virtual_Mall='E-MALL  
4 B')  
5 -----
```

6 First, we'll retrieve all database objects for E-MALL B from the dynamic table of Fig. 13  
7 (rows 4 through 9 of the ID column). Second, we'll retrieve all database objects for the virtual  
8 part, which is E-SHOP A-2 (rows 1,2, and 3 of the ID column of Fig. 13).

9 So far, we have explained a virtual e-mall where a single database exists for all dynamic e-  
10 malls, e-shops, e-distributors and web sites, also, only one database table for the virtual ones.  
11 This is just one arrangement, and as we'll see. The dynamic database table can be located at  
12 different servers in a network or over the Internet, or a combination of both.

13 Fig. 15 shows four servers and each having a database table. For simplicity, we are  
14 showing only the table's ID and they represent the table's ID of Fig. 13 (for dynamic servers)  
15 and the table's ID of Fig. 14 (for the virtual server). For example, **SERVER A 1510** hosts  
16 **EMALL A 1512**. The IDs for the ID column 1512 are 1,2 and 3 (ID column of Fig. 13).  
17 Next, the e-mall name at the column *Mall\_Name* is "E-Mall A" for rows 1,2 and 3 (ID column

1 of Fig. 13). Fig. 15 shows a virtual server 1500 with a virtual table 1502 (ID column of Fig.  
2 14). **VIRTUAL SERVER** 1500 sends request and receives data 1511 from dynamic  
3 **SERVER A** 1510 which hosts **E-MALL A** 1512; sends request and receives data 1521 from  
4 **SERVER B** 1520 which hosts **E-MALL B** 1522, and sends request and receives data 1531  
5 from **SERVER C** 1530 which hosts **E-MALL C** 1532.

6 Requests are sent and data received from different servers in the network or over the  
7 Internet. And they are requests for database objects (table rows) from each server. Once  
8 they're received, they are combined and a single dynamic table is formed, then it is related with  
9 the virtual table 1502 (ID column) at virtual server 1500. Finally, the result is presented as a  
10 single database table to a user. It should be noted that a single virtual database table 1502 (ID  
11 column) is shown at virtual server 1500, but it can be more than one table at a single server,  
12 or it can be, more than one database tables at multiple servers. Also, virtual server 1500 can  
13 have a dynamic database table and have e-commerce and/or e-services as well, although not  
14 shown.

15 Also, any of the dynamic servers can offer services (e-services/contents) and make them  
16 available to the virtual server. For instance, if **SERVER A** 1510 were offering services  
17 instead. Services from **SERVER A** 1510 would be available to **VIRTUAL SERVER** 1500 and  
18 to all its e-malls, satellite e-malls, e-distributors, e-shops and web site. **VIRTUAL SERVER**

1 1500 will make all the interfacing with end-users and then pass any user's entered information  
2 to SERVER A 1510 and also save at its database as required.

3 We've described a virtual network system where a resource infrastructure exists (VNRI)  
4 and it is offered to third parties (e-mall) and these third parties offer this infrastructure to  
5 others (e-mall, e-distributors/manufacturers). It is the intent of this invention to allow this  
6 same method to be used as virtual solutions where a single infrastructure exists and it is used  
7 by more than one business and each business will in turn uses it and offer solutions to its  
8 clients. For example, a manufacturer uses VNRI and offer solutions to its supplier. The  
9 manufacturer is like an e-mall and its suppliers are like e-shops. In other words, the process  
10 involves three levels: first level is VNRI; second level is an e-mall and the third level is an e-  
11 shop. So, VNRI (first level) offers its solution to various locations (second level) and each  
12 location (second level) offers VNRI solution (first level) to at least one other location (third  
13 level). Also, instead of e-malls, it can be companies with e-portals, also instead of e-shops it  
14 can e-services from a company's departments.

15

16

17 II) ON-LINE INTERFACE TRANSLATION METHOD

1       As it has been explained, each web page has objects. They can be audio, video, images,  
2       links, forms, text, etc. We'll be discussing text object in particular. As we know, the virtual  
3       server e-mall will be used worldwide. Furthermore, it must be able to display *text objects* in  
4       the user's own native language. Also, it should have means of translating *database objects*  
5       (products/services), and also, e-shops and e-malls interfaces (text objects).

6       First, let's explore the on-line page interface translation (text objects); second, the  
7       translation of products/services (database objects), and lastly, we'll show how they will  
8       enhance the user's virtual shopping experience.

9       The following are the steps for on-line interface translation: a user at a client selects a  
10      drop down or any other means for selecting a foreign language and the client sends this  
11      request to a server. A program in the server receives the user selected foreign language as  
12      **encoded information** and it represents the user selected foreign language. The program in the  
13      server has a function to translate text objects and it also has program code means for  
14      translating database text objects.

15      Let's first explore the text object translation. The function translating text objects receives  
16      two parameters: one is the **encoded information** (the user selected foreign language) and the  
17      other parameter is a **code reference**. The **code reference** is used to search a database table  
18      and retrieve a row with text objects in different foreign languages in each of its columns. The

1    **encoded information** is used for retrieving a column which is a reference to the database  
2    table's row that was received in the previous step and it contains text object is in the user  
3    selected foreign language.

4        Now let's explore the translation of database text objects. The **encoded information**  
5    (selected foreign language) is combined with a SQL text string and it is a query expression.  
6    Once a program in the virtual server executes the query expression, the query's result will only  
7    include database text object in the user selected foreign language.

8        We'll be using ASP (Active Server Page) technology along with JavaScript language for  
9    our explanation, since, they are frequently used on the Internet. Fig. 16 shows a table  
10   (languageTable) with four columns: **ID**, **Phrase\_Code**, **ENG** and **POR**. Fig. 17 shows two  
11   parts of a drop down form object and, Fig. 18 the actual HTML code for the drop down form  
12   object of Fig. 17.

13        Let's explore the drop down form object (Fig. 17). It has two parts: the element that is  
14   hidden from the user and, the element used by a user to change its state (make a selection).  
15   The table of Fig. 17 has two columns: the first column (Hidden Elements), represents the  
16   hidden part that the browser uses to send a representation of the user selection to server; the  
17   second column (drop down choices), shows the drop down choices used by the user to make  
18   a selection. Fig. 18 is the HTML code that creates the drop down of Fig. 17.



1       Back to Fig. 18. The line "<form name=changelanguage action=newlanguage.asp>" (line  
2 # 2) indicates the start of a form object and "</form>" (line # 7) the end of it. The value at the  
3 **action** element of the form tag is "**newlanguage.asp**" (line # 2). It will be the page that the  
4 web browser will request from the virtual server once the form **changelangue** is submitted  
5 and pass its object (form objects) to it as well. The line "<select name=language  
6 onChange='changeLanguage()>" (line # 3) indicates the start of a drop down object and  
7 "</select>" (line # 6) the end of it. The following two lines are the drop down options. The  
8 line "<option value=ENG>English</option>" (line # 4) is the first option and "<option  
9 value=POR>Portuguese</option>" (line # 5) is the second option. And they are the choices  
10 "**English**" and "**Portuguese**" of the drop down displayed on the web browser.

11       Let's say that the drop down currently has the option **English**, as the selected one. Next,  
12 a user changes it to **Portuguese**. Three things will happen: first, the browser will transfer  
13 control to *function changeLanguage()*; (line # 10) second, the function will submit the form  
14 to the virtual server *document.changelanguage.submit()* (line # 12) - it is the value at the  
15 **action** of the form **changelanguage** "<form name=changelanguage  
16 action=newlanguage.asp>" (line # 2); and third, the server will retrieve *newlanguage.asp*  
17 (Fig. 19) and pass **POR** as a parameter to it - **POR** represents the user's selection.

1        Fig. 19 shows the **newlanguage.asp** (ASP page at the virtual server) with a JavaScript  
2        program. It is divided into two parts: from beginning of the file (line # 1) up to line # 27 is the  
3        ASP program executed by the server. Next, code from the <HTML> (line # 28) tag all the  
4        way to the end of the file (line # 45) is the portion sent back to the user's web browser  
5        (HTML objects). Also, codes after the <HTML> tag that are surrounded by "<%" and "%>"  
6        are pieces of ASP code (it usually returns a value and it is inserted into the HTML page).

7        Let's delve into **newlanguage.asp** file Fig. 19. The first line "var  
8        Language=Request.Form("language")" (line # 1) requests the drop down user's selection. It  
9        was passed to the server by the client computer (web browser). Now, the variable **sLanguage**  
10       has the string value "**POR**" (the user selected language).

11       Next, we have a function called "doTranslate(sPhraseCode,sChosenLanguage)" (line # 3).  
12       It has two parameters: *sPhraseCode* and *sChosenLanguage*. The first parameter,  
13       **sPhraseCode** receives a **code reference** value to be used to search the column **Phrase\_Code**  
14       of the **languageTable** (Fig. 16). The second parameter, **sChosenLanguage** will be an  
15       **encoded information** and it is the user's selected language (the value "**POR**" at the variable  
16       **sLanguage**). Now the function *doTranslate()* will open connections (database and record set)  
17       then perform a query at the **languageTable** (Fig. 16). Next, it retrieves a value from column  
18       **POR** - the user's selected language stored at the variable **sLanguage**. In our example the

1 value at the Query variable is *"SELECT POR FROM languageTable WHERE*  
2 *Phrase\_Code='ENGL'"* (line # 16 and line # 17). The last line is  
3 *return(languageRS(sChosenLanguage))* (line # 20) and it returns the retrieved value from the  
4 data table.

5 The next set of code of Fig. 19 is a partial HTML code within the ASP page. It includes  
6 the actual drop down and the ASP code (between "<%" and "%>" - line # 32 and line # 35)  
7 inserts the function *doTranslate()* returned value into the HTML page. There are two lines  
8 with *Response.Write()*'s. The first one "<% Response.Write(doTranslate("ENGL",  
9 sLanguage)) %>" (line # 32) will translate element for the drop down that says "Inglês", and  
10 the second one "<% Response.Write(doTranslate("PORT", sLanguage)) %>" (line # 35) will  
11 say "Português". The new drop down is in Portuguese, which is the selected language.

12 Let's explore the previous explanation. Lets take the first drop down element "<%  
13 Response.Write(doTranslate("ENGL", sLanguage)) %>" (line # 32). The function  
14 *doTranslate()* receives two parameters: *sPhraseCode* and *sChosenLanguage*. *SphraseCode*  
15 receives the value "ENGL" and *sChosenLanguage* receives the value "POR" (user's selected  
16 language). Function *doTranslate()* searches the column **Phrase\_Code** of the **languageTable**  
17 (Fig. 16) for the value "ENGL" (row # 7 of the ID column), then, retrieves the value at the  
18 column **POR** (user selected language) and it is "Inglês".

1       The function *doTranslate()* is very basic. Other means will be used for retrieving  
2       information instead of opening and closing databases and record sets. It was presented as is  
3       for simplicity.

4       Now that we know how text objects of the interface gets translated, we'll go one step  
5       further and translate the actual database objects and see how they relate to the virtual e-malls,  
6       satellite e-malls, e-shops, e-distributors and web sites.

7       Fig. 20 shows a database table with two rows and four columns: **ID**, **Product\_ID**,  
8       **Language** and **Product\_Title**. The **ID** column holds the table ID's for each row. The  
9       **Product\_ID** column holds the ID for each product. The **Language** column, holds the  
10      language code for each product in the table. The **Product\_Title** column holds the title for  
11      each product in the language specified by the language code at the column **Language**. The  
12      first row (row # 1 of the ID column) is in Portuguese (the value at the **Language** column is  
13      "**POR**" for Portuguese) and the second row (row # 2 of the ID column) is in English (the  
14      value at the **Language** column is "**ENG**" for English).

15      Now lets say that, a user views an e-shop and selects a language other than the one  
16      currently displayed by the web browser on the client's computer screen. The new screen will  
17      show a drop down with the new language, and also, all the web page text objects translated  
18      accordingly. This process is called web page text object translation.

1       Let's now explain how to translate the database objects that are displayed on the page  
2 (e.g. product's title, description, etc.). Once a new language is selected, all the database  
3 objects at the new web page must be in the new language. And because of this, the new query  
4 will also be affected. Some products/services might be in a different foreign language other  
5 than the selected one. As we can see, the number of available products/services will also  
6 change.

7       Fig. 21 shows a dynamic table (DynamicTable2) and it is the same table of Fig. 13 except,  
8 we've deleted one column **Product\_ID**, and added two new ones: **Product\_Title** and  
9 **Language**. Following, we have the SQL from our previous example. Let's include the user's  
10 selected language in it.

11 -----  
12 SELECT \* FROM DynamicTable2 WHERE Language='POR' AND ( Mall\_Name='E-  
13 MALL B' OR Shop\_Name IN (SELECT Virtual\_Type\_Name FROM VirutalTable WHERE  
14 Virtual\_Mall='E-MALL B') )

15 -----

1       As we see, the query's result will only retrieve rows 1, 4 and 7 of the ID column (Fig. 21).  
2       In our previous SQL example without the use of the user selected language, rows 1 though 9  
3       of the ID column of Fig. 13 were retrieved.

4       As it can be seen, the user's selected language affects the virtual e-malls, satellite e-malls,  
5       e-shops, e-distributors and web sites by changing the number of available good/services. For  
6       example, there are eight products in English (rows 2,3,5,6,8,9,11 and 12 of the ID column)  
7       and four in Portuguese (rows 1,4,7 and 10 of the ID column).

8

### 9       III) TEMPLATE UPLOADING METHOD

10       There are two ways that a front-end can be interfaced with VNRI's back end: first by  
11       selecting a pre-set interface that is within VNRI, second, by a user uploading a customized  
12       one. In either case, the interface is the front-end means for accessing the VNRI e-commerce  
13       back-end means.

14       In the case of a user-uploaded interface, it will have means for guiding a server in the  
15       inserting of program code and other pieces of information as well (include files). After the  
16       uploaded interface is received and processed by the server a new web sited is created and as  
17       we've said before, it is the front-end means for a client accessing the server back-end means.

1        Customized user interface will allow a user at any time to change the front-end look  
2        without programming knowledge or knowledge of the server back-end structure. Also, a  
3        customized user interface will allow the inclusion of e-commerce in a web site without the  
4        expense or expertise that an e-commerce infrastructure requires.

5        For example, a customized interface is the same one that is in use by a web site. This new  
6        customized interface is now uploaded to a server with means for processing and creating a  
7        new interface. Now, a client is able to fetch both and present an interface with similar  
8        appearance to a user. The web site will present its services in the same way as it had done  
9        before and the server where the customized interface is located will present the e-commerce  
10       part, thus allowing, a web site without e-commerce means to include e-commerce easily and  
11       affordably.

12       We already know that, a client computer communicates with a server computer over a  
13       TCP Internet connection. A client sends requests to server then the server sends objects in the  
14       HTML format back to client. These objects are then displayed in the client computer as web  
15       pages. Fig. 22 shows this setup. Client 2230 sends request 2231 to server 2232 than server  
16       2232 retrieves an ASP page and *include files* (files to be included into the ASP page) as  
17       needed 2233. Server 2232 then sends the page back to client 2230 as HTML page 2234.

1       A true virtual e-mall, satellite e-mall, e-shop or web site will also need means for the user  
2       to change its interface without having to access the server computer which hosts them. Fig. 23  
3       shows an ASP page template. The first part 2370 creates variables and opens database  
4       connection. After a connections is opened, indexes representing links to products/services at  
5       the e-mall, e-shop or web site is created 2372. After a link is selected, a new page with the  
6       products /service information 2371 (e.g. title, description, image, etc.) is received. Finally,  
7       before leaving the page, opened connection is released to free resources that are no longer  
8       needed 2373.

9       As it was shown at Fig. 22. An ASP page has *include files* 2233 and, it can be a program  
10      in any acceptable ASP language. First, these *include files* are inserted into the ASP page, then  
11      the server executes them. Their results will in turn be inserted into the new created file. This  
12      new file will then be transmitted to the requesting computer (client) in the HTML format  
13      2234.

14      Fig. 24 shows the same template from Fig. 23 with *include file tags* inserted into it.  
15      Before.inc 2480 is where database connection, record set and variables are created.  
16      Indexes.inc 2482 create web page links and they will be used to select a new a web page -  
17      main\_page.inc 2481. At the end of the page, all page's objects are released - after.inc 2483.



1        So, a HTML page can be created from an ASP template by having *include files* inserted  
2        into it. And once we have a standard ASP templates format, we can exchange them in and out  
3        without affecting *include files*. All we need to do is: create a new template with a new look;  
4        insert the *include files* tags, next, a new web page is created based on the same set of *include*  
5        *files*. Changes can also be made directly to an *include file* without affecting the remaining  
6        ASP page.

7        Since we are creating virtual e-malls, e-shops, e-distributors and web sites, users will need  
8        means for exchanging interfaces without having physical access to the virtual server. It can be  
9        a user supplied interface or one supplied by the virtual server.

10       Fig. 25 shows how to replace an interface. The top part 2590 is the old interface and the  
11       bottom part 2591 is the new one. Users can upload a new interface template, or chose one  
12       from the virtual server 2592. The new interface 2591 will then replace 2593 the old interface  
13       2590.

14       All interfaces reside in folders within the virtual server. Let's explain what a folder is  
15       before proceeding any further. A Folder is a container of files or other folders. A good  
16       example is a file cabinet. A file cabinet has drawers - they are like folders; files within a drawer  
17       are like files within a folder.

1        Fig. 26 shows two ASP templates. The first one ShopCart.asp 2600, has a folder's path  
2        added to each include file "<!-- #include file="ShopCart/before.inc" -->"; the second template  
3        ShopPage.asp 2601, has a different path in the include file "<!-- #include  
4        file="ShopPage/before.inc" --> ". The folder's path as part of the *include files* tells the server  
5        where *include files* are located. For example, "<!-- #include file="ShopPage/before.inc" -->"  
6        tells the server that **before.inc** is located in the folder **ShopPage**.

7        Fig. 27 shows the folder structure for Fig. 26. It has three folders: **ShopCart** 2702, stores  
8        all include files for template ShopCart.asp 2600 (Fig. 26); **ShopPage** 2703 stores all include  
9        files for template ShopPage.asp 2601 (Fig. 26), **Templates** 2704 stores ASP template  
10       ShopCart.asp 2600 and ShopPage.asp 2601 (Fig. 26).

11       Let's delve into folder **Templates** 2704. As we see, the file **ShopCart.asp** 2705 is the  
12       same template 2600 (Fig. 26) and the file **ShopPage.asp** 2706 is the same template 2601 (Fig.  
13       26). Once a new template is uploaded with its respective include files and include file's path  
14       into the **Template** folder 2704, a new interface will then be in place for an e-mall, e-satellite e-  
15       mall, e-shop or web site.

16       Furthermore, a single ASP template can be uploaded and have a program in the server to  
17       create others with the correct path in each one of them. We've shown a very simple folders  
18       structure with only three folders, in reality, a folder structured can have any number of folders,

1 and a folder can have any number of files. Also, we've named the folder as ROOT 2707 (Fig.  
2 27), although it can be any name. It could've been a name for an e-mall, satellite e-mall, e-  
3 shop or web site as well.

4 Fig. 28 shows a setup where a client computer with a web browser 2813 receives  
5 templates from Virtual Server 2815 and Host Server 2814. Host Server 2814 establishes a  
6 communication link 2810 with Virtual Server 2815 and uploads a set of templates 2820-a.  
7 When the web site is first accessed. Client 2813 requests and receives a new interface 2820-a  
8 from Host Server 2814 through connection 2811. Once a user selects any link that refers to  
9 the virtual shopping environment, or to virtual web pages (located at Virtual Server 2815).  
10 Client 2813 then requests and receives the next set of interfaces 2820-b (web pages) from  
11 Virtual Server 2815 through connection 2812. From this point on, Virtual Server 2815 will  
12 supply the new interface 2820-b and it will have the same look as the interface 2820-a located  
13 at the host computer 2814. The user at the Client Computer 2813 will only notice one  
14 interface 2820-c, displayed on the computer's screen, although, it is coming from two different  
15 servers or two different URL's.

16  
17 As presented, templates can be uploaded to the virtual server and be used by an e-mall,  
18 satellite e-mall, e-shop or web site for the purposed of presenting a single interface to a user at  
19 a single web browser location, although it may be fetched from multiple locations.

#### IV) FILE WITH SETTINGS UPLOADING METHOD

A user uploaded file with settings contain parameters that are associated with a variable name or any other means for allowing its processing and the extraction of its parameters. For example, the file with settings may have the following: "*background\_color='ffffff'*", in this case, once the parameter *'ffffff'* is extracted from the file with settings, the program will know that it represent a value to be used to set the web site or a web page's background color to white.

Now we'll discuss the uploading of file with settings that goes along with an interface. Its purpose is to allow an interface to have more than one appearance. It may have different background colors, different text colors, different text sizes, different table formats, etc.

Fig. 29 shows a file with settings 2900 being uploaded to a server with templates 2902 and at the server it is read and its contents included in a template 2904.

Fig. 29a shows the same file structure of Fig. 27 except two more files are present at templates folder 2984: ShopPage.stg 2980 (file with settings for template ShopPage.asp) and ShopCart.stg 2982 (file with settings for template ShopCart.asp).

Fig 30 shows a file with settings and its settings being used by a template file (ShopPage.asp) 3041. It has **Page\_Background=#ffff** 3040 it is the color white for the template's background (web page) and it is represented in the <BODY> tag as **BGCOLOR="#ffff"** 3052. Also, **Page\_Font\_Color=#000000** 3042 is **TEXT="#000000"** 3054 (default black color for all text in the web page). The **Page\_Link\_Color=#0000ff** 3044 is the blue color used by the web page links and it is **LINK="#0000ff"** 3056.

There are three more settings and they are used for other purposes than changing the pages default colors. They are used to change text part of the web page created by the web server. **Title\_Font\_Color=#ffff00** 3046 is **color="#ffff00"** 3058 at the <FONT> tag and it is the color yellow. **Title\_Font\_Size=3** 3048 is **SIZE="3"** 3060 and **Title\_Font\_Face="Verdana, Arial"** 3050 is **FACE="Verdana, Arial"** 3062. The last three settings will make "THIS IS A PAGE TITLE" 3064 in yellow, with the font size of "3" and "Verdana" or "Arial" for the font type.

1        It is now clear that all that is needed for changing a web page template is to upload a file  
2        with settings with different settings and the web page will change its look accordingly. This  
3        method will allow a single template to be remotely programmed and have more than one look  
4        without changing the template or its content.

5  
6        Functions necessary to read the file with settings, separate and place its settings in a  
7        memory array is not shown and is done as is for sake of simplicity. Also, the file with settings  
8        has just a few settings values, there can be any number and be used differently for the purpose  
9        of changing the interface appearance.

## 10 11 12    V) SATELLITE E-MALLS

13  
14        A satellite e-mall is very similar to an e-mall, except it is a loosen one by making. It has  
15        one or more e-services, also, there can have many satellite e-malls and each offering all kind of  
16        e-services and making them available to any e-mall or e-shop in the VNRI. It is up to an e-  
17        mall to include or exclude a satellite e-mall.

18

1        Once an e-mall includes one, the e-mall inherits its e-services and its behaviors as well. An  
2 e-mall with a satellite e-mall will be able to present the satellite e-mall's e-services along with  
3 its e-shops products and most of the cases it will be specialized e-services. For example: an e-  
4 mall includes a satellite e-mall that is specialized in car sales. Every time users access the e-  
5 mall and the satellite e-mall's e-service that is part of it. The e-mall e-commerce part may only  
6 display products that are related to cars in general.

7  
8        Also, the e-service from a satellite e-mall has guiding means for guiding the e-mall in the  
9 displaying of its products along with the e-service. The guiding means may be an **encoded**  
10 **information** in a hidden field of a form or it may be words within the e-service page.

11  
12        There are other means for presenting e-services to a satellite e-mall as well. And it is  
13 using e-services located in a different server than a server within the VNRI. The server will  
14 make its e-services available to VNRI and VNRI will make them available to a satellite e-mall.  
15 E-malls at VNRI are able to incorporate the satellite e-mall and its e-services by incorporating  
16 the satellite e-mall containing them.

17  
18        As we've mentioned before, HTML is displayed on a computer screen as objects. Each e-  
19 service is one or more object and they can be text, image, form, links etc., and at least one

1 object in the page will have at least one **encoded information** and the **encoded information**  
2 will be the guiding means for guiding the server to include at least one other object in the page  
3 and transmit it to a client.

4  
5 As it has been presented so far. The virtual network has e-malls and e-malls has e-shops  
6 and web sites. Also, as we'll see, the virtual network has means for creating and maintaining  
7 satellite e-malls and make them available to all e-malls in the virtual network (pre-set satellite  
8 e-malls). Also, the same means for creating and maintaining satellite e-malls is available for  
9 each e-mall in the network (customized satellite e-mall).

10  
11 Satellite e-malls at the virtual network are specialized and e-malls use them as is and will  
12 not be able to make changes to them. On the other hand, a satellite e-mall created by the e-  
13 mall is customized and can be changed at will by the e-mall that created it.

14  
15 A satellite e-mall is an e-mall where e-services and e-commerce can be offered  
16 simultaneously at the same web page and e-services will enhance the e-commerce counterpart.  
17 If a user at a web browser is viewing a satellite e-mall web page and he/she selects a link, drop  
18 down list or any other means at the e-service side. The e-commerce part will change as well to  
19 reflect the e-services.



1  
2 E-services offered by a satellite e-mall can be of any kind: Auction, Car Sales, Realty, etc.  
3 Also, it can be of any kind of specialized web site: Sports, News, Weather etc.  
4

5 The virtual network will have many e-services available to e-malls, satellite e-malls, e-  
6 shops, e-distributors and web sites. And any will be able to select and include web pages of e-  
7 services provided for by the virtual network. Each of these e-services web pages will be  
8 already translated into the foreign languages that are supported by the virtual network. These  
9 e-services web pages will permit customization process without having e-services of their  
10 own.  
11

12 The e-commerce and the e-services may or may not reside at the same location. They can  
13 be at a single or multiple URL addresses, folders, databases or database tables.  
14

15 Let us explore this concept a little further. Let us say that that an e-mall has some e-  
16 shops selling sporting goods, some others selling cars accessories and still others selling music  
17 CD's (let us call it: **e-mall A**). Now let us say that a satellite e-mall in the virtual network is  
18 specialized in offering car services (let us call it: **satellite e-mall Car Sales**). Besides its

1 specialization in car services, this satellite e-mall will have departments related to cars, for  
2 instance, a department for car accessories.

3  
4 Now, if the **e-mall A** includes the **satellite e-mall Car Sales** into its architecture. Every  
5 time a user access the **satellite e-mall Car Sales** he/she will be able to view all information  
6 about cars and also view all car accessories products from the **e-mall A** along with car  
7 accessories products from distributors/manufacturers.

8  
9 The idea is to allow e-commerce and e-services to be displayed on a single web page  
10 although they come from two different locations. In the above example, the e-commerce  
11 comes from **e-mall A** and the e-services comes from the **satellite e-mall Car Sales** that is part  
12 of the virtual network and available to all e-malls in the virtual network system. The satellite  
13 e-mall will allow an e-mall to offer more specialized services than it could otherwise. Anyone  
14 skilled in the art knows that the two separate contents having relationships and displayed at  
15 the same page can be both e-services or they can be both e-commerce.

16  
17 Now is time for us to delve a little further and discuss how the satellite e-mall will  
18 enhance the shopping experience in an e-mall

19

Fig. 31 shows this arrangement, the virtual network 3100 has three satellite e-malls 3102, 3104 and 3106 and each of this satellite e-malls having distinct services. **The satellite e-mall A 3102 specializes in Car Sales, satellite e-mall B 3104 specializes in Auction and satellite e-mall C 3106 specializes in Sports.** Also, there are two e-malls: **e-mall A 3108 and e-mall B 3112.** Now each one will have more services and products for selling besides the ones that are available within their respective e-shops.

**E-mall A 3108** now has **Car Sales 3102** and **Auction 3104**. When a user views an Auction for a specific product (e-services) the drop down tree at the web page on the left (e-commerce) will be set accordingly to show the user that a similar product is also available in an e-store within the e-mall. The same is true for **e-mall B 3112**. It has a **satellite e-mall B 3104 (Auction)** and **satellite e-mall C 3106 (Sports)**. Now **e-mall B 3112** will have **Auction** and also sell **Sports** related products from its e-shops along with **Sports** related products from e-distributors/e-manufacturers.

Since the virtual network will host a variety of e-malls and they in turn will host a variety of e-shops. Any e-mall with a **satellite e-mall Auction** will offer to their e-shops means for placing their products in the auction and it will be available to every e-mall hosting the **satellite e-mall Auction**.

1  
2 If we look just bellow **e-mall A 3108** and **e-mall B 3112**, we'll see that each e-mall has a  
3 customized satellite e-mall. **Customized satellite e-mall A 3110** for **e-mall A 3108** and  
4 **customized satellite e-mall B 3114** for **e-mall B 3112**.  
5

6 The customized satellite e-mall is different than a satellite e-mall supplied by the virtual.  
7 network. The e-mall creates and maintains a customized satellite e-mall and supplies its e-  
8 services or select one from the virtual network, also, selects departments for it's e-commerce.  
9 A customized satellite e-mall is unique to each e-mall while the satellite e-mall supplied by the  
10 virtual network is the same for any e-mall that offers its services.  
11

12 Fig. 31a and Fig. 31b are further overview of how e-services can be incorporated into a  
13 satellite e-mall through VNRI. E-services (contents) are received from third parties (servers)  
14 by the VNRI and made available to satellite e-malls as if each e-services had been provided by  
15 the VNRI. Each e-service displayed on a client computer screen is one or more objects and  
16 they can be image, text, form, frames, etc.  
17

18 Fig. 31a shows e-services provided by third parties 3140 to VNRI 3142. As shown,  
19 Satellite e-mall A 3144 and Satellite e-mall B 3146 are receiving e-services available at VNRI

1 3142, although they are located at the third party server 3140. Now that satellite e-malls have  
2 these e-services and after an e-mail incorporates any satellite e-mail, all e-services that are part  
3 of the satellite e-mail become part of the e-mail.

4  
5 Fig. 31b shows another method of making e-services available to satellite e-malls. As  
6 shown, there are three e-services: E-service A 3160, E-service B 3162 and E-service C 3164  
7 and all three are made available to VNRI 3166, and they are supplied to VNRI by a third party  
8 server. Now VNRI 3166 will make all three e-services available to satellite e-malls. Satellite  
9 e-mail A 3168 has E-service A, E-service B, E-service C and contents from the VNRI (3170).  
10 Satellite e-mail B 3172 has E-service A and E-service B (3174). As it has been previously  
11 explained, once an e-mail incorporates either satellite e-mail A or satellite e-mail B, their e-  
12 services along with VNRI's contents (if any) will be part of the incorporating e-mail.

13  
14 As we turn to fig. 31c it shows **satellite e-mail A 3182 with a group of contents 3180**  
15 **(Content A, Content B and Content C). Satellite e-mail A 3182 is incorporated by E-mail A**  
16 **3184 and E-mail A 3184 incorporates "Content C" and "Content G" content page 3186.**  
17 **"Content C" and "Content G" are presented virtually by E-mail A 3184 since "Content C" is**  
18 **retrieved from Satellite e-mail A 3182 content group 3180 and "Content G" from VNRI**  
19 **3196 content group 3198. Arrow lines A1 and A2 show these relationships. The rest of the**

1 figure is self-explanatory and anyone skilled in the art will be able to understand it's meaning  
2 based on the explanation just given. Figs. 31b and 31c are just one arrangement that can be  
3 accomplished with the combination of Satellite e-malls, E-malls and the virtual nature of  
4 VNRI.

5  
6 Once an e-service is displayed on a client computer it will be one or more objects and at  
7 least one of these objects will enable communication between the client computer and VNRI.  
8 The client computer will process the object and initiate a communication with VNRI passing  
9 parameters about the displayed content page and the server's location from where it was  
10 fetched. Next, VNRI will use the received parameters than it will fetch the e-service/content  
11 (the same one that is displayed at the client computer) from the server where it is hosted.  
12 After receiving the e-service/content VNRI will process its objects and fetch at least one  
13 other of its stored e-service/content or e-commerce that is related to the received content,  
14 and return it to the client computer. Therefore, e-service/content displayed at the client  
15 computer will be fetched from the VNRI and at least one other server and both contents  
16 having a relationship with each other. Also, instead of requesting the e-service/content from  
17 the server every time an end user at the client computer requests it, VNRI may fetch it once,  
18 process it, and stores the processed objects for future use. Also, it can be fetched before its

1 first use by using VNRI's registration means described at the "CUSTOMIZED WEB SITES  
2 (CONTENTS)" section.

3  
4 As we turn to fig. 31d it illustrates what we've just described, and it shows **content**  
5 **page** located at a **remote computer** 3170-d and this remote computer is not part of the  
6 VNRI infrastructure. Also, a **group of contents** 3178-d are located at the VNRI and a  
7 **single content page** 3172-d having contents from both sources. Content 3170-d is  
8 requested and received (arrow 3176-d) by a client computer (single page 3172-d) from the  
9 remote computer, than at least one content will be requested and received from VNRI  
10 (arrow 3174-d). Once we review the contents received from **remote computer** 3170-d and  
11 part of **content page** 3172-d it says: "This content is from here" and the contents from  
12 VNRI 3178-d, part of **content page** 3172-d say: "This"; "Content" and "From". As we  
13 analyze, **content page** 3172-d has the word "This" and it is related to the inserted word  
14 "This" from content group 3178-d, also, content 3172-d has the word "content" and related  
15 to the word "Content" from content group 3178-d. This illustration shows that at least one  
16 object (words in this illustration) must have a relationship between both content sources and  
17 it doesn't necessarily need to be an exactly match for the relationship, nor the related words  
18 need to be present at the final page displayed at the client computer screen. Their  
19 relationship may only happen at the indexing mechanism with VNRI.

1       As it has already been explained, objects of content 3170-d are used as the input means  
2       (searching keys) for locating other contents at VNRI and returning at least one matching  
3       content to the requesting client computer and the client computer will display the content  
4       received from the remote computer 3170-d and at least one content received from VNRI  
5       3178-d. Also, the content can be fetched as requested for the first time and stored for future  
6       use, or, the contents can be fetched before its first use and it can be done by using VNRI's  
7       registration means described at the "CUSTOMIZED WEB SITES (CONTENTS)" section.

8  
9       Now, lets turn to fig. 31e and it shows one object for inserting inline content into a web  
10      page, other means exist as well. We'll view three techniques that can be used, although others can  
11      be devised and used as well without departing from the true scope of the invention. The first one  
12      3184-e shows '`<iframe src="www.vnri.com/default.asp?id=remoteabc&location=www.remote-  
13      computer.com" width="100%"></iframe>`'. The "src" indicates the source from where the client  
14      computer's browser will fetch the next content from, VNRI in our example. As the client  
15      computer contact "www.vnri.com" it will request the web page "default.asp" and pass  
16      " id=remoteabc&location=www.remote-computer.com" to the web page. The web page  
17      "default.com" will retrieve the value "remoteabcd" from "id", and the value "www.remote-  
18      computer.com" from "location". Next, the value "remoteabc" of the "id" parameter is used to  
19      search the content 3170-d stored at VNRI and use at least one of its objects to fetch at least one



1 content 3178-d that are stored at VNRI. Finally, "location" has the URL location of the remote  
2 computer where VNRI will fetch the content 3170-d from, if this is the first time that it has been  
3 requested.

4  
5 3186-e is just another way of fetching contents from VNRI and as we look at the object's  
6 parameters, instead of "location" it has "words" and a list of words that can be passed to VNRI.  
7 In this case, instead of VNRI fetching the content from the remote computer, it uses the passed  
8 words for searching contents. And the last one, 3188-e shows just the "id" of the remote  
9 computer and it can be used to fetch objects from the remote computer that are already stored at  
10 VNRI. Any one skilled in the art will know that the presented arrangements are not the only ones  
11 and many others can be devised and implemented as well.

12  
13 Also, contents can be fetched by the VNRI from third party servers (third party servers not  
14 associated with the VNRI infrastructure) and pass them on to other third part servers. The other  
15 third party servers in turn will make the contents received from VNRI available to end users at  
16 client computers accessing them. So, a end user at a client computer accesses a server and  
17 request at least one content, the server requests the at least one content from VNRI. Next, VNRI  
18 will request the at least one content from other third party servers and return any received content  
19 from the requested third party servers, then transmit them to the third party server that initiated

1 the request. VNRI acts as an intermediary between at least two third party servers and performs  
2 the job of fetching and supplying contents from/to other servers.

3  
4 Fig. 31f illustrates what we've just described. VNRI 3194-f fetching contents from one or  
5 more sources, from two computer in this arrangement, 3190-f and 3192-f and making any  
6 content fetched from the remote sources available to another computer 3196-f and the computer  
7 3196-f connected to a client computer 3198-f and the content 3199-f that was fetched from  
8 3192-f displayed at the client 3198-f.

9  
10 Fig. 32 shows what we've discussed so far. On the left part of the web page 3220 is  
11 where the e-commerce part will be displayed and on the right part 3222 is for the e-services.

12  
13 Fig. 33 shows a further embodiment of Fig. 32. On the left links 3320 has a tree drop  
14 down menu with **Shoes** and **Apparel** for the **departments**. The **department Shoes** has  
15 **Women** as **category** and the **category Women** has **Tennis** as a **sub-category**. On the right  
16 window 3322 (satellite e-mail activity window) has three drops downs: **department 3328**,  
17 **category 3330** and **sub-category 3332**. At the top of each drop down there is the user choice  
18 for it. **Shoes** for **department 3328**, **Women** for **category 3330** and **Tennis** for **sub-category**

1 3332. We're showing drop downs as means for the communication between the e-services  
2 3322 and the commerce 3320.

3  
4 Other means can be used as well, like, hidden form elements with department, category  
5 and sub-category codes encoded in them. Also, embedded information with key words at the  
6 e-service web page, or even, the use of the information at the web page as indexing key words  
7 that the server will use to search the e-mall's database and provide a list that match the words  
8 in the e-service web page. One other way to embed information in a page is by using a pair of  
9 HTML tags. Opening tag "<!--" and closing tag "-->". The browser will not display any  
10 content in between these two tags. The purpose of having content embedded within the page's  
11 content is to allow the insertion of words and/or information that are not yet part of the page's  
12 content. It may be words with synonymous meaning about other words on the page or related  
13 words that will help in further describing the page's content. Also, there are other kinds of  
14 tags that are used just for the purpose of embedding content to the HTML page and they are  
15 familiar to anyone with skill in the art.

16  
17 In the case of customized e-services. It can have embedded information pertaining to the  
18 e-mall supplying the customized e-service, and, whenever the e-service is used virtually by  
19 another e-mall, satellite e-mall or web sited in the virtual network. Also, products from e-

1 stores of the e-mall that is supplying the e-service will be available virtually at each one based  
2 on the embedded information at the e-service's web page.

3  
4 Fig. 34 is a further embodiment of Fig. 33. It shows a list of products displayed on the  
5 right window 3322 that is the user's selection (Tennis 3321) from the left window 3320. There  
6 are two products with respective images, titles with links and a check box. **Tennis** as the  
7 selected **sub-category** 3321 has product **Tennis Shoe ABC** (image 3444, title with a link  
8 3442 and a check box 3440) and **Tennis Shoe XYZ** (image 3434, title with a link 3436 and a  
9 check box 3438). Also, a button 3446 is present on right window 3322 for the purpose of  
10 placing selected items in a shopping cart. In our example both items will be placed in the  
11 shopping cart, since they are both checked (check box 3440 and check box 3438).

12  
13 As we turn to fig. 34a it is a further embodiment of fig. 34 and fig. 31d and it shows a  
14 single content page having two e-services (contents). **Content** 3460 and another **group of**  
15 **contents** 3462 and both are located within VNRI. **Content** 3460 can be hosted/stored along  
16 with or separated from the **content group** 3462. Now the **content group** 3462 might be  
17 related to the same category as **content** 3460, or they can be from a different category, since  
18 the objects at **content** 3460 will direct VNRI which contents from the **content group** 3462 to  
19 fetch and include into the **content page** 3464. It should be understood that previously visited

1 contents' objects (they are explained at "CUSTOMIZED WEB SITES (CONTENTS)") could  
2 be used as searching key (input objects) as well. They can be used alone or in conjunction  
3 with the objects from **content** 3460. The idea is that at least one object of the **content being**  
4 **fetched** 3460 or objects of previously visited content(s) be used as input for searching other  
5 contents located at VNRI, **content group** 3462. Also, the **final page** 3464 is very simplistic  
6 for sake of clarity and not intended to obscure this invention. Contents from VNRI 3462 and  
7 inserted into **content page** 3464 can be a complete content page, a brief description of the  
8 actual content, partial content with links pointing to the location where the content originated,  
9 etc. In case a link is present and once it is clicked the user will be directed to the content's  
10 source and view the actual content, and, it can be within VNRI or leave VNRI altogether.

11

12

## 13 VI) SURF USER LIST

14

15 As we've seen so far, the virtual network comprises of e-malls, satellite e-malls, e-shops,  
16 e-distributors/e-manufacturers and web sites. Also, an e-shop from one e-mall can sell  
17 virtually in another e-mall. As we see, a web browser user accessing an e-mall in the virtual  
18 network will come across web pages from different e-shops dynamically located at the e-mall,

1 web pages from virtual e-shops (e-shops dynamically located at a different e-mall) and also  
2 from e-distributors/e-manufacturers.

3  
4 Lets say that the web browser user later return to the e-mall and want to find a product  
5 that he/she saw on a previous visit. Lets also say that the product was from a virtual e-shop  
6 and it is no longer available virtually at the e-mall. The user will think that the item is no  
7 longer available because the e-shop is no longer virtually located at the e-mall. From this  
8 scenario it is clear that there is a need for a mechanism to track and keep the user surfing  
9 experience.

10  
11 Each page of information supplied to each client have a **surf code reference** and it is used  
12 for automatically storing a reference for each information supplied to each client and it forms  
13 the **surf user-list**. Once the user requests his/her surf user-list, the server will use each **surf**  
14 **code reference** and create the **surf user-list** and sent it to the user. A **surf user-list** will only  
15 include information that was previously viewed by the user.

16  
17 Also, an end-user at a client with a login means and after logging on the server, he/she  
18 will be able to request the server to save his/her surf user-list for later review. The server will

1 save the surf user-list based on the end-user ID that is part of the login information in the  
2 server.

3  
4 Fig. 35 shows the communication that takes place between a web server and a web  
5 browser, when the web browser accesses the web server for the first time. The web browser  
6 3552 initiates communication 3553 with the web server 3554. The web server 3554 generates  
7 a tracking number 3560 and send it 3556 to the web browser 3552, now the web browser  
8 3552 will store it in a cookie 3566. This process will allow the web server 3554 to keep track  
9 of the web browser that is accessing it. We're showing the ASP technology but it can be  
10 implemented by other technologies as well.

11  
12 Fig. 36 shows a web browser 3552 accessing web pages from: e-malls, e-stores and web  
13 sites. Web browser 3552 fetches web pages from each one of them. From e-shop A 3672 end-  
14 user at the web browser 3552 - view **products 123 and 456** (3674). From e-shop B 3676 will  
15 view **product 789** (3678). From e-mall A 3680 - view **product 145** (3682). From e-mall B  
16 3684 - view **products 100 and 102** (3686) and from web site 3688 views web pages **News**  
17 and **Weather** (3690).

18

1        Fig. 37 shows the process that takes place at each request. The web browser 3552 sends  
2        a request 3702 to the web server 3554, the web server 3554 send a request for cookie called  
3        **user\_tracking\_code** 3722 to the web browser 3552. Now the web browser 3552 will retrieve  
4        the value "0123656" from the cookie **user\_tracking\_code** 3566 and send it 3706 to the  
5        server 3554. The web server 3554 will first save the requested web page or the product page's  
6        code in the session variable **user\_tracking\_code** 3560 and second it will fetch the web page  
7        or the product's page 3726 and sent it 3728 to the web browser 3552.

8  
9        Fig. 38 shows the process that takes place once a user at the web browser 3552 requests  
10       to view his/her surf list 3560. The web browser 3552 sends a request 3810 to the web server  
11       3554. The web server 3554 will send a request 3812 to the web browser 3552 for the value of  
12       the cookie **user\_tracking\_code** 3566 and the web browser 3552 will fetch the value  
13       "0123656" from it and send 3814 to the web server 3554. The web server 3554 will fetch the  
14       values that are stored in the session variable **user\_tracking\_code** 3560. Next, the web server  
15       3554 uses the list just retrieved from the session variable 3560 and searches the database  
16       3832. And finally, it will fetch web pages and/or product's page that correspond to the values  
17       in the session variable 3560 and sent the page to the web browser 3552 through connection  
18       3816.



Fig. 39 shows a web page with links on the left 3950 and they are links for the user surf list. They are the same ones that are stored in the session variable **user\_tracking\_code** 3560 (Fig.38). On the right 3952 we see a page that the user has just selected from his/her surfing list 3950. And the selection was 123 (3951) and its contents are now displayed on the right 3952 and they are the item's image 3958, the item's title and description 3956 and also a button 3954 to place it in a shopping cart. Once again, a very basic web page was shown for sake of simplicity.

We've shown values stored in a session variable **user\_tracking\_code** 3560 (Fig.38). It can be stored in a database as well for the purposed of tracking and recording the user activity and always have his/her surf list available.

## VII) CUSTOMIZED WEB SITES (CONTENTS)

Besides the user surfing tracking system, the virtual network will also have means for customizing web sites to each user. Once a user is registered, personal information and personal preferences will be used as to allow the virtual network to customize web sites to each user as to reflect the user's preferences and also the user's location.

1

2       There will be a plurality of information about a specific subject matter and upon a user  
3       access the server it will customize a web site. A customized web site will allow a server means  
4       for presenting different information about the same subject matter tailored uniquely to each  
5       user based on each user's pre-set information stored in the server.

6

7       A user furnishes information to a server and once the same user returns to the server and  
8       log in, the server will only supply information to the user that is relevant to the user and based  
9       on the user pre-set information that is stored in the server.

10

11       Let's say that user has **Soccer** and **Football** for sports preferences and the local of  
12       residency is San Francisco, California, USA. Another one residing in Salvador, Bahia, Brazil  
13       has **Soccer** and **Basketball** for sports preferences. Now, both users will access the same web  
14       site at the same time and view sports. The user residing in San Francisco, California, USA will  
15       view information for Soccer and Football for leagues from San Francisco, California and  
16       USA, while the second user will view sports information for leagues from Salvador, Bahia and  
17       Brazil.

18

1       The same method can be used to set user previous visited web pages or products page.  
2       For instance, a user visits a product page about fishing. Next time he/she comes to any web  
3       site within VNRI with e-services, fishing related information will be part of the web site along  
4       with fishing products (e-commerce part).

5  
6       As we turn to fig. 40 it shows what we've described so far. VNRI 4000 has user's  
7       preferences 4018 and for "User A" they are "Sports" and "USA" (USA could have been  
8       extracted from the user's registration information) and once "User A" access the virtual  
9       network 4000 "User A" will view only sports related contents from USA, 4004 and 406 at  
10      client computer 4002, and they are contents 4008 and 4014 respectively. Once "User A"  
11      returns to VNRI the same two contents will be displayed automatically as it is shown by fig.  
12      41.

13  
14      It is to be understood that all e-commerce (goods/products) and e-services  
15      (articles/services) within the virtual network are available to all e-malls, satellite e-malls, e-  
16      shops, e-distributors and web sites virtually. Also, means will be available as to allow each e-  
17      mall to make its customized e-service(s) available to others e-malls, satellite e-malls, e-shops,  
18      e-distributors and web sites at the virtual network.

19

1        Many more arrangements can be created and incorporated in this invention. While it has  
2        been fully described in connection with the illustrated embodiments, it will be appreciated and  
3        understood that modifications may be made without departing from the true spirit and scope  
4        of the invention. We've used the term e-malls and e-shops all along but instead of e-mall, it  
5        can be called e-portal and instead of e-shops, it can be a combination of e-shops and e-  
6        services. Also, the arrangements presented can be used individually or in any combination  
7        thereof. Furthermore, this invention will allow the creation of a worldwide virtual  
8        environment.

9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19